Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 83 (2016) 695 – 699

Procedia Computer Science

The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016)

Multiagent Hybrid Architecture for Collaborative Exchanges between Communicating Vehicles in an Urban Context

Laurent Lucien[a,c,*], Christophe Lang[a], Nicolas Marilleau[b], Laurent Philippe[a]

[a] *UMR CNRS 6174 Femto-ST/DISC, University of Science and Technology, Besançon, France*
[b] *UMI 209 UMMISCO, IRD/UPMC, Bondy, France*
[c] *PSA Peugeot Citroën - Digital, Data & Connectivity Engineering, France*

Abstract

Nowadays, we are more and more surrounded by powerful and intelligent communicating objects. Many of these objects, as smartphones, watches, detectors and soon cars, are moving in increasingly interconnected environments, have abilities to communicate with each other and to exchange information. A collaborative approach allows these entities to exchange information and objectives and to implement rules in a structured manner in order to optimize the execution of their own mission and, therefore, the operation of the system in general. For example, collaborative behaviours and informations exchanges could improve the movement of vehicles in an urban center and avoid traffic jams. Our contribution puts a stress on a definition of collaboration in the context of mobile communicating entities. For the sake of agent-based modeling, we also list challenges raised like technical architecture and data organisation. Then we propose an hybrid architecture for collaborative exchanges with an example based on communicating vehicles in an urban context and implemented on the GAMA platform.

Peer-review under responsibility of the Conference Program Chairs

Keywords: Multiagent Systems ; Hybrid Architecture ; Collaboration ; Communicating Vehicles

1. Introduction

Smart objects (computers, smart phones, etc) are now occupying a wide part in our life and are fully interconnected. They intend to carry on services for an individual user or for the community. In the case of communicating vehicles, interaction between vehicules is currently mainly done through an external infrastructure (cloud). In such infrastructure, a vehicle is considered as client that sends data to the infrastructure and receives specific information extracted by the infrastructure from gathered data. This interaction scheme does not however match all vehicle needs and vehicle-to-vehicle interactions are needed to improve security and services to drivers for instance. This interaction scheme does not give a response to novel vehicle requirement, especially in the domains of security and driver assis-

* Corresponding author. +33-3-81-66-20-68
E-mail address: laurentolivier.lucien@gmail.com

Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2016.04.154

tance. What does it happen if the infrastructure is down or unavailable (in a tunel for instance)? Vehicle-to-vehicle interactions could give a response to that.

According to some studies [1,2,3], there is no real autonomy if the vehicle can not fully cooperate or collaborate with any other entities. These same studies attempt to describe how these intelligent and autonomous vehicles will change our lives through three main goals: guarantee the road safety, improve the quality of life, permit the accessibility for all. So collaboration is the key of the success of such applications.

In the multiagent systems literature, several works tackle the collaboration domain [4,5,6]. Due to the variety of mobile smart objects, a large variety of collaboration scheme exists but exchanges between connected objects are often specific to the application domain. We can note a lack of domain-expert oriented methods providing concepts and tools to qualify and study the collaboration in a complex system. Thus, we tackle the problematic of collaboration and promote methods, concepts and tools to qualify exchanges between mobile entities (vehicles, drones, etc) evolving in a complex environment (city, forest). Introducing and describing collaboration between mobile entities should enhance their journey and their efficiency. For that, we take advantage of agent based systems for which the versatility allows to describe real systems by interactions between autonomous entities. In this paper, we present an hybrid agent architecture that can be used to model collaborative exchanges between mobile entities and its assessment as a autonomous vehicle model in a multi-agent system.

In section 2 of this article, we first propose a definition of the concept of collaboration and, according to this definition, we present technical challenges to implement a collaborative behaviour. In the third section we propose a collaborative agent architecture, the data organisation and the world representation. Then, in section 3, we propose a new hybrid agent architecture with its main communication components, in an urban context. At last, a first implementation is presented using GAMA platform in section 4.

2. Collaboration and Multiagent System

In this section, a general analysis on collaboration is given as a preambule of a short overview about communication between agents. It permits to refine our point of view on collaboration and outline linked issues in the domain of multi-agent system.

In the domain of Multi-Agent Systems, cooperation and collaboration concepts highlight interactions between agents and cognition: it needs some coordination actions and conflict resolution algorithms to achieve tasks [7]. Nevertheless there are differences between collaboration which is "*a form of interaction who is interested in how to distribute the work among several agents, whether it is centralized or distributed technics*" and cooperation that "*remains the prerogative of beings capable of having an explicit project therefore cognitive agents.*" [8] Collaboration is thus considered as cooperation refined by the development of a mutual understanding associated with a shared point of view of the task being solved by several interacting individuals [9,10]. In the context of mobile objects like communicating vehicles, collaboration intends to achieve an individual mobility objective while performing a collective local task, by exchanging information between two or more mobiles. It is an intentional and cognitive process: it also results from the wishes of each mobile which collaborates with an effort of sharing selected information and a common vision of the goal to be reached.

Several architectures have been proposed to permit communication between agents or collaborative processes such as, for example: (i) the Belief-Desire-Intention (BDI) architecture of Rao and Georgeff [11], (ii) Touring Machines of Ferguson [12], (iii) InteRRaP model of Müller [13], etc.

All these previous models propose a multi-layer architecture with a layer for the world representation, a layer for basic behaviours, another one for planned behaviours and a final one generally dedicated for communication and/or collaborative process. Communications between agents are often limited to this dedicated layer (the higher one, the more "cognitive" layer which initiate communication if needed). This approach implies that agent do everything it can before asking help to another agent, somewhere in the simulation environment. But in cases where agents represent quick mobile objects, this consideration could be useless because the processing time would be too long. For example, if a vehicle detects an accident in front of it, it must inform immediatly vehicles all around to avoid another one.

So, for *modeling collaborative exchanges between mobile entities*, an hybrid agent architecture is required to answer quickly to environmental stimuli (reactive part). This architecture must also include some storage capacity to record experiments and an management of objectives and priorities (deliberative or cognitive part). It must be

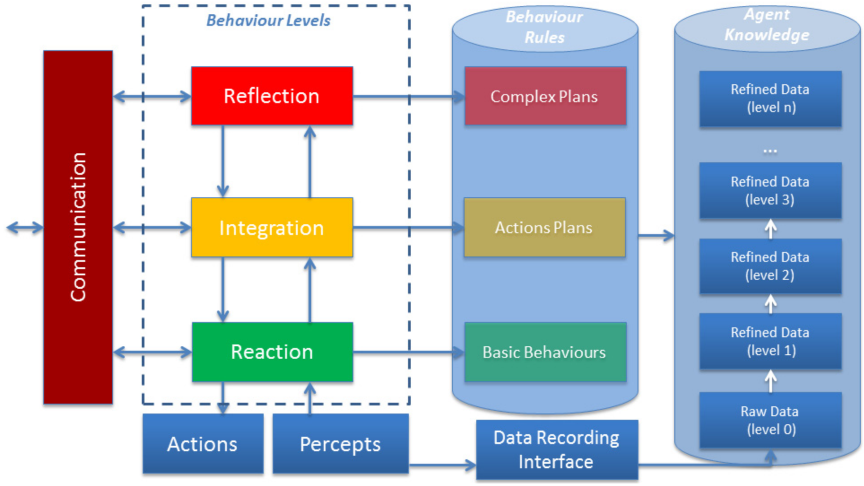


Fig. 1. A Hybrid Architecture of communicating agent composed of multi-level knowledge, behaviour rules, organised around sensors, effectors and communication media

associated with a definition of its interactions with the model: the way to record information (world representation, goals, interactions between agents) and the communication protocol to use with its dependencies that depend on the multiagent development platform.

3. Collaborative Agent Architecture

In order to model a collaborative multi-agent system, we propose in this section an hybrid agent architecture based on three layers (see Fig. 1). This achitecture addresses the issues presented in the previous section. The proposition is a layered architecture where each layer is dedicated to a specific level of cognition. These layers are projected on a library of behaviour rules and the agent knowledge hierarchical database. Low level data are collected through interfaces and refined to be delivered to the analysis algorithms and data storages. The architecture also includes a communication module that implements direct inter-agent communications for a matter of flexibility and efficiency.

The architecture lower level is actually its interface to the environment. It is composed of two modules: the *percepts module* and the *action module*. The *percepts module* gathers information from the environment and dispatches it to both behaviour and data recording parts of the architecture. In case of a vehicle, the perception module gathers data from all the electronic sensors. The *action module* transmits orders to the various effectors of the agent. For a vehicle, this includes the management of electronic boxes that are mandatory for the link with car bodies. This lower level of the architecture also includes a raw data recording interface that sends data to the agent knowledge hierarchical database. This interface does not include an intelligent program. Data are not filtered nor sorted at this level. The purpose of this module is to facilitate the registration of raw data without the need for a specific program described in an upper layer. We explain below how data is organized and how it is used.

Now we describe *behaviour levels*. The first layer (*reaction*) manages all reactive behaviours. This is the operational module where all basics reactions are treated. The second layer (*integration*) brings some more refined behaviours with a deepening of the decision process. This is the tactical module where actions plans with special constraints are implemented. The third layer (*reflection*) allows to integrate complex behaviours based on the previous layers decision process. This is the strategic module with some long-term views and where complex missions to be accomplished are implemented. The different layers are requested by message or by program (this may vary depending on the implementation, we will discuss about this in the next section). The three layers use the *behaviour rules* library and the *agent knowledge* database.

The *behaviour rules* library is a hierarchical object. The atomic bricks are the *basic behaviours* of agents, their reactions to the environment perception. Each *basic behaviour* has its own preconditions, constraints and postconditions. Then we find the *actions plans* which are composed by *basic behaviours*. Finally there are *complex plans* composed by *actions plans*. For all actions or plans, we can also find preconditions and constraints. All these *basic behaviours*, *actions plans* and *complex plans* are linked together. The organisation of these links are like a tree of actions. Each *basic behaviour*, *actions plan* or *complex plan* is called by a specific layer that is a kind of entry point in the action tree. This leads to a basic work can be conditionned by the *reaction layer* and possibly continue the whole plan by triggering the top layer.

The main objective of the *agent knowledge* database is to correctly structure data to manage them quickly and communicate them to the outside. For example, sensors collect raw information that will be stored in the knowledge base. Then this raw information is analyzed and treated gradually by actions attached to the different layers. We can attach other information to complete and refine basic information.

For us, we can define some collaborative actions and communications at different level. So the *communication module* is an important part of this architecture. To ensure a high level of responsiveness, this module is transverse. The *communication module* can be called by any layer, selects the right protocol depending on the geographical distance and the assigned task (diffusion, partial collaboration, full collaboration) and maintains links with the listener as long as the mission requires, so lets move from one protocol to another as needed. This block allows monitoring of the communication process between agents.

4. Instanciation in an Urban Context

This case study intends to test and evaluate our collaborative agent architecture. For that a real case study was choosen: modeling smart cars moving in a small town. These vehicles must circulate from their home to their workplace as quick as possible and avoid traffic jams. In an urban context and to model vehicles, we find here the driving component and all about basic detection cases like an accident or stopped vehicles on the road. Agents *Vehicle* are defined with multiple properties: *standard* vehicles, *smart* vehicles (with a capacity for observation and analysis of the world), *communicating* vehicles (with communication capability) and *connected* to a centralized entity like a *Cloud*. Several *basic behaviours* are identified: (i) determine its status, (ii) move into the environment from point A to point B, (iii) observe the environment (road traffic) if it is an intelligent vehicle. Some *action plans* are also set: (i) identify a traffic jam, (ii) identify a stalled car, (iii) evaluate a new route to avoid a blocked road, (iv) diffuse information to other agents, (v) diffuse information to a centralized entity (private cloud or mutualised infrastructure).

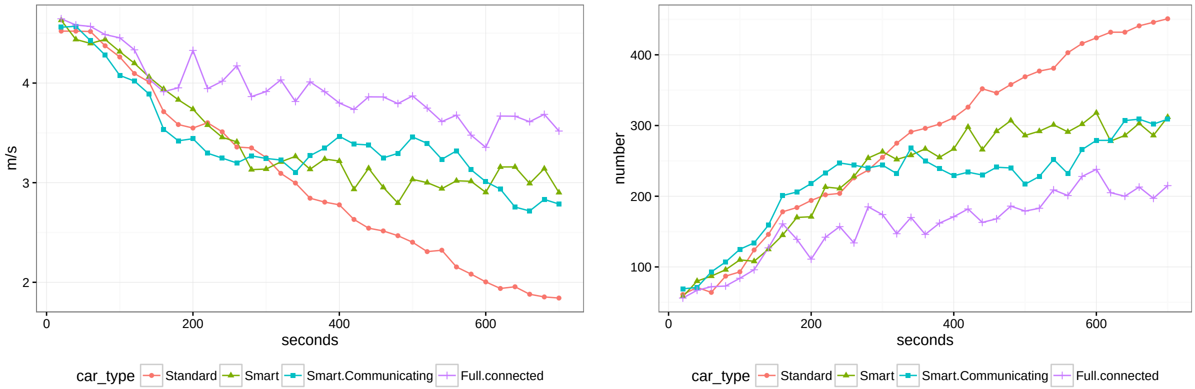
To implement the collaborative agent architecture in an urban context, we chose the GAMA platform because it supports modeling and provides a simulation development environment for building spatially explicit agent-based simulations. The GIS functions are also implemented natively.[14,15] We use advanced driving skill[16] to avoid the driving management.

During the simulation, we generate some random events (like an accident or a brake down) to create some traffic jams. In order to demonstrate the effectiveness of different types of vehicles, we have chosen two indicators: the average speed of vehicles in the environment and the number of stopped vehicles. We performed several simulations with 700 vehicles in an urban area to assess different types of vehicles and their impact on the traffic.

The results of the simulation are shown on the figure 2. We can note that the average speed is significantly higher when vehicles begin to exchange information on traffic trends to enable dynamic change of path. While standard vehicles have a constant average speed degradation, communicating vehicles are able to maintain an average speed approximately constant on this sample. We also find that fully connected vehicle (with exchange of supplementary information with a centralized entity) maintain a better average speed. In correlation, the number of stopped vehicles tends to remain stable for communicating vehicles. Again, fully connected vehicles are more frequently in motion.

5. Conclusion

In this paper, we defined the collaboration between mobile agents. According to this definition and the urban context, we proposed a multiagent hybrid architecture with three dedicated layers (reaction, integration and reflection). These layers are projected on a library of behaviours rules adapted to all situations (emergency, consolidation, distribution, etc) and an agent knowledge. To illustrate the communicating vehicles problematic, a first implementation has been realized on GAMA platform.



(a) Average speed (b) Number of stopped vehicles

Fig. 2. Simulation with 700 agents

We will continue the development of the collaborative agent architecture by implementing the full multigraph concept and by improving the information dissemination process in connection with the communication block. We will establish more elaborate action plans in connection with road traffic to show the interest of structured communication between the communicating vehicles and smart infrastructure. BDI mechanisms will be added to organize the internal decision processes and allow the creation of joint plans between agents. Finally FIPA performatives will be evaluated to ensure their relevance in a context of quick exchanges of information in a constrained environment.

References

1. G. Silberg, R. Wallace, G. Matuszak, J. Plessers, C. Brower, D. Subramanian, Self-driving cars: The next revolution, White paper, KPMG LLP & Center of Automotive Research, 2012.
2. R. Hudda, C. Kelly, G. Long, J. Luo, A. Pandit, D. Phillips, L. Sheet, I. Sidhu, Self driving cars, College of Engineering University of California, Berkeley, Berkeley: College of Engineering University of California, 2013.
3. B. Flury-Hérard, H. De Tréglodé, Les véhicules communicants nécessitent-ils de nouvelles réglementations, 2015.
URL `http://www.ladocumentationfrancaise.fr/var/storage/rapports-publics/154000746.pdf`
4. B. J. Grosz, Collaborative Systems (AAAI-94 Presidential Address), AI magazine 17 (2) (1996) 67.
5. S. K. Lo, A collaborative multi-agent message transmission mechanism in intelligent transportation system–a smart freeway example, Information Sciences 184 (1) (2012) 246–265.
6. M. Zhang, Q. Bai, F. Ren, J. Fulcher, Collaborative Agents for Complex Problems Solving, in: C. Mumford, L. Jain (Eds.), Computational Intelligence, Vol. 1 of Intelligent Systems Reference Library, Springer Berlin Heidelberg, 2009, pp. 361–399.
7. N. R. Jennings, Controlling cooperative problem solving in industrial multi-agent systems using joint intentions, Artificial Intelligence 75 (2) (1995) 195 – 240. doi:http://dx.doi.org/10.1016/0004-3702(94)00020-2.
URL `http://www.sciencedirect.com/science/article/pii/0004370294000202`
8. J. Ferber, Les systèmes multi-agents : vers une intelligence collective, Informatique, Intelligence Artificielle, InterÉditions, 1995.
9. G. Weiss, Multiagent systems: a modern approach to distributed artificial intelligence, MIT press, 1999.
10. M.-F. Blanquet, Web collaboratif, web coopératif, web 2.0.: quelles interrogations pour l'enseignant documentaliste, Formation des personnes ressources en documentation.
11. A. S. Rao, M. P. Georgeff, BDI Agents: From Theory to Practice, in: IN PROCEEDINGS OF THE FIRST INTERNATIONAL CONFERENCE ON MULTI-AGENT SYSTEMS (ICMAS-95, 1995, pp. 312–319.
12. I. A. Ferguson, Touring Machines: Autonomous Agents with Attitudes, Computer 25 (5) (1992) 51–55. doi:10.1109/2.144395.
URL `http://dx.doi.org/10.1109/2.144395`
13. J. P. Müller, M. Pischel, The Agent Architecture InteRRaP: Concept and Application, Tech. rep. (1993).
14. A. Drogoul, E. Amouroux, P. Caillou, B. Gaudou, A. Grignard, N. Marilleau, P. Taillandier, M. Vavasseur, D.-A. Vo, J.-D. Zucker, Gama: multi-level and complex environment for agent-based models and simulations, in: Proceedings of the 2013 international conf. on Autonomous agents and multi-agent systems, International Foundation for Autonomous Agents and Multiagent Systems, 2013, pp. 1361–1362.
15. P. Taillandier, D.-A. Vo, E. Amouroux, A. Drogoul, Gama: a simulation platform that integrates geographical information data, agent-based modeling and multi-scale control, in: Principles and Practice of Multi-Agent Systems, Springer Berlin Heidelberg, 2012, pp. 242–258.
16. P. Taillandier, Traffic simulation with the gama platform, in: International Workshop on Agents in Traffic and Transportation, 2014, pp. 8–p.